

FACULTY OF TECHNOLOGY AND ENGINEERING

**DEVANG PATEL INSTITUTE OF ADVANCE
TECHNOLOGY AND RESEARCH**

DEPARTMENT OF COMPUTER ENGINEERING

A.Y. 2023-24 [EVEN]

LAB MANUAL

CE266: SOFTWARE ENGINEERING

ID: 22DCE006-Probin Bhagchandani
22DCE011-Nisarg Chaudhari
22DCE018-Urva Dave
22DCE040-Japan Kachhiya

PRACTICAL-9

AIM: Design the Test Suites and Test Cases for the given project Box Testing, White Box Testing, Gray Box testing.[Manual and Automated Testing]

Use tool: Selenium Automation (Web driver, TestNG)

Installation of Selenium automated testing tool and Create Test Scripts in Selenium.

Reference link: <https://www.guru99.com/seleniumpython.html>

To design test suites and test cases for the Inviq – Smart Inventory Management System project using various testing methodologies and Selenium for automated testing, let's break it down into Black Box Testing, White Box Testing, and Gray Box Testing along with manual and automated testing.

1. Black Box Testing (Functional Testing):

Black Box Testing focuses on the functionality of the system without considering its internal structure. Here, we'll design test cases based on the system's requirements.

Test Suite: Functional Test

SuiteTest Cases:

1. Verify adding inventory item:
 - a. Add a new inventory item with valid details, ensure it is added successfully.
 - b. Try to add an inventory item with missing or invalid details, ensure proper validation.
2. Verify editing inventory item:
 - a. Edit an existing inventory item's details, ensure changes are saved correctly.
 - b. Try to edit an inventory item with invalid details, ensure proper error handling.
3. Verify deleting inventory item:
 - a. Delete an existing inventory item, ensure it is removed from the system.
 - b. Try to delete a non-existing inventory item, ensure proper error message is displayed.

4. Verify managing inventory levels:
 - a. Increase inventory quantity for an item, ensure the quantity is updated correctly.
 - b. Decrease inventory quantity for an item, ensure the quantity is updated correctly.
5. Verify generating bills/invoices:
 - a. Generate a bill/invoice for a set of inventory items, ensure it is accurate.
 - b. Generate a bill/invoice for an empty set of inventory items, ensure it is handled properly.
6. Verify tracking sales:
 - a. Record a sale of inventory items, ensure the sale is reflected in the system.
 - b. Try to record a sale without specifying inventory items, ensure it is not allowed.
7. Verify setting reorder points:
 - a. Set a reorder point for an inventory item, ensure it is saved correctly.
 - b. Try to set a reorder point for a non-existing inventory item, ensure it is not allowed.
8. Verify generating reports:
 - a. Generate a report of inventory items based on various criteria (e.g., category, quantity), ensure it is accurate.
 - b. Try to generate a report with incorrect or invalid criteria, ensure it is handled properly.

2. White Box Testing (Structural Testing):

White Box Testing involves testing the internal structure of the system, including code paths and logic.

Test Suite: Structural Test

SuiteTest Cases:

1. Test inventory item authentication:
 - a. Test the method to add an inventory item with valid credentials, ensure it is successful.
 - b. Test the method to add an inventory item with invalid credentials, ensure it fails gracefully.

2. Test data validation:

- a. Test inventory item form validation, ensure all required fields are validated properly.
- b. Test input validation for managing inventory levels, ensure proper error messages are displayed.

3. Test database operations:

- a. Test adding a new inventory item record to the database, ensure it's inserted correctly.
- b. Test updating inventory item records in the database, ensure changes are persisted.

4. Test application security:

- a. Test role-based access control for inventory management functions, ensure only authorized users can perform actions like adding, editing, or deleting inventory items.
- b. Test session management, ensure sessions expire after a certain period of inactivity.

These white box test cases focus on the internal operations of the inventory management system, ensuring that the code behaves as expected and that the system is secure and reliable.

3. Gray Box Testing:

Gray Box Testing combines elements of both Black Box and White Box Testing. Testers have partial knowledge of the system's internal workings.

Test Suite: Gray Box Test

Suite Test Cases:

1. Test integration with a database management system (DBMS):
 - a. Verify that inventory data is synchronized correctly between the system and the DBMS.
 - b. Check for proper error handling in case of DBMS connection failures.
 - c. Verify that database transactions are properly managed and rolled back in case of failures.
2. Test integration with an email service for sending notifications:
 - a. Verify that email notifications for low inventory levels, successful transactions, or other events are sent successfully.
 - b. Check for proper error handling if the email service is unavailable, ensuring that notifications are queued and sent once the service is back online.
3. Test performance under load:
 - a. Simulate a large number of users adding, editing, or deleting inventory items simultaneously to assess system responsiveness.
 - b. Monitor system resources (CPU, memory, etc.) to ensure there is no performance degradation under load.
 - c. Conduct stress testing by gradually increasing the load to determine the system's maximum capacity and identify any bottlenecks.
4. Test error handling and recovery:
 - a. Intentionally trigger errors such as invalid input, database connection failures, or server timeouts, and verify that the system handles them gracefully.
 - b. Verify that error messages are clear and informative, and that appropriate logs are generated for troubleshooting.
 - c. Test recovery mechanisms by simulating a system failure and verifying that the system can recover data and resume operations without data loss.

5. Test security features:

- a. Verify that sensitive data such as user credentials and inventory information are stored securely and protected from unauthorized access.
- b. Test authentication and authorization mechanisms to ensure that only authorized users can access sensitive functionality and data.
- c. Conduct penetration testing to identify and address potential security vulnerabilities.

Manual Testing: Perform exploratory testing, usability testing, and acceptance testing to validate the system's behavior from an end-user perspective.

Automated Testing with Selenium:

Utilize Selenium WebDriver and TestNG to automate test cases for UI interactions and functional testing.

CONCLUSION: By adopting this approach, you can create comprehensive test suites and test cases that cover a wide range of functionalities and scenarios in the Inviq Inventory Management System. These tests can address aspects such as inventory item management, billing and invoicing, sales tracking, report generation, and integration with accounting software. They can be executed manually by testers or automatically using tools like Selenium WebDriver and TestNG, ensuring thorough and efficient testing of the system.

Staff Signature:

Grade:

Remarks by the Staff:

